CLAIMS

What is claimed is:

1. A method for managing addition and deletion of network nodes from and to a secure multicast or broadcast group of nodes in a communications network without a single point of failure, wherein each of the nodes is associated with one of a plurality of replicated group controllers and wherein the nodes and the group controllers are logically organized in a binary tree that represents the network nodes and the group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the group, intermediate nodes represent other network nodes, and root nodes represent the group controllers, the method comprising the steps of:

joining one of the group controllers to the plurality of replicated group controllers in a local network;

establishing, by one of the group controllers, a secure communication channel between one of the group controllers and another of the group controllers using a key exchange protocol;

receiving a request to add or delete a node of the group from a load balancer that is coupled to the plurality of group controllers;

creating and storing a new group session key for each node in each branch of the tree that is affected by adding or deleting the node from the group;

distributing a group session key from one of the group controllers to the network nodes.

2. A method as recited in Claim 1, wherein distributing a group session key further comprises:

receiving a token value at the group controller to designate the group controller as having permission to selectively generate the group session key and to generate node keys associated with the intermediate nodes and the leaf nodes; and

creating and storing the group session key only when the group controller has the token value.

3. A method as recited in Claim 1, wherein distributing a group session key further comprises:
determining whether the secure multicast or broadcast group has a node that is leaving the secure multicast or broadcast group;
determining which of the intermediate nodes are affected by the leaving node;
updating keys associated with the affected intermediate nodes;
generating a new group session key; and
sending the new group session key to the leaf nodes.

4. A method as recited in Claim 3, wherein updating keys comprises:
generating a new key of a parent node of the leaving node; and
encrypting the new key of the parent node with a key of a node adjacent to the parent node.

5. A method as recited in Claim 1, wherein distributing a group session key further comprises:
receiving a request message from one of the plurality of nodes to join the secure multicast or broadcast group;
determining which of the intermediate nodes are affected by the joining node;
updating keys associated with the affected intermediate nodes;
generating a new group session key and a private key of the joining node; and
sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

6. A method as recited in Claim 5, wherein updating keys comprises performing a one way hash function on the keys associated with the affected intermediate nodes.

7. A method as recited in Claim 1, wherein receiving a request comprises receiving the request at a load balancer having a single virtual address that represents the plurality of group controllers.

8. A method as recited in Claim 7, further comprising the step of load balancing network traffic that is directed from a plurality of the nodes to the plurality of group controllers.

9. A method as recited in Claim 1, wherein establishing a secure communication channel comprises exchanging a public key of the group controller with all other group controllers in the plurality of replicated group controllers based upon optimized broadcast Diffie-Hellman protocol.

10. A method as recited in Claim 5, wherein establishing a secure communication channel comprises:
receiving a public key value that is broadcast by the joining node;
sending a collective public key value from the nodes to the joining node;
computing a shared secret key;
creating and storing a group shared secret key by exchanging private key values.

11. A computer-readable medium comprising one or more sequences of instructions for managing addition and deletion of network nodes from and to a secure multicast or broadcast group of nodes in a communications network without a single point of failure, wherein each of the nodes is associated with one of a plurality of replicated group controllers and wherein the nodes and the group controllers are logically organized in a binary tree that represents the network nodes and the group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the group,

intermediate nodes represent other network nodes, and root nodes represent the group controllers, and which instructions, when executed by one or more processors, cause the processors to carry out the steps of:

joining one of the group controllers to the plurality of replicated group controllers in a local network;

establishing, by one of the group controllers, a secure communication channel between one of the group controllers and another of the group controllers using a public key exchange protocol;

receiving a request to add or delete a node of the group from a load balancer that is coupled to the plurality of group controllers;

creating and storing a new group session key for each node in each branch of the tree that is affected by adding or deleting the node from the group;

distributing a group session key from one of the group controllers to the network nodes.

12. A computer-readable medium as recited in Claim 11, wherein distributing a group session key further comprises:

receiving a token value at the group controller to designate the group controller as having permission to selectively generate the group session key and to generate node keys associated with the intermediate nodes and the leaf nodes; and

creating and storing the group session key only when the group controller has the token value.

13. A computer-readable medium as recited in Claim 11, wherein distributing a group session key further comprises:

determining whether the secure multicast or broadcast group has a node that is leaving the secure multicast or broadcast group;

determining which of the intermediate nodes are affected by the leaving node;

updating keys associated with the affected intermediate nodes;

generating a new group session key; and

sending the new group session key to the leaf nodes.

14. A computer-readable medium as recited in Claim 3, wherein updating keys comprises:
generating a new key of a parent node of the leaving node; and
encrypting the new key of the parent node with a key of a node adjacent to the parent node.

15. A computer-readable medium as recited in Claim 11, wherein distributing a group session key further comprises:
receiving a request message from one of the plurality of nodes to join the secure multicast or broadcast group;
determining which of the intermediate nodes are affected by the joining node;
updating keys associated with the affected intermediate nodes;
generating a new group session key and a private key of the joining node; and
sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

16. A computer-readable medium as recited in Claim 15, wherein updating keys comprises performing a one way hash function on the keys associated with the affected intermediate nodes.

17. A computer-readable medium as recited in Claim 11, wherein receiving a request comprises receiving the request at a load balancer having a single virtual address that represents the plurality of group controllers.

18. A computer-readable medium as recited in Claim 17, further comprising the step of load balancing network traffic that is directed from a plurality of the nodes to the plurality of group controllers.

19. A computer-readable medium as recited in Claim 11, wherein establishing a secure communication channel comprises exchanging a public key of the group controller with all other group controllers in the plurality of replicated group controllers based upon Diffie-Hellman protocol.

20. A computer-readable medium as recited in Claim 15, wherein establishing a secure communication channel comprises:
receiving a public key value that is broadcast by the joining node;
sending a collective public key value from the nodes to the joining node;
computing a shared secret key;
creating and storing a group shared secret key by exchanging private key values.

21. A method of managing addition and deletion of network nodes from and to a secure multicast or broadcast group of nodes in a communications network, wherein each of the nodes is associated with a first group controller comprising information that is replicated in a plurality of group controllers, and wherein the nodes and the group controllers are logically organized in a binary tree that represents the network nodes and the group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the group, intermediate nodes represent other network nodes, and root nodes represent the group controllers, the method comprising the steps of:
joining the first group controller in a local network in which the plurality of group controllers are coupled;
establishing a secure channel between the first group controller and the plurality of group controllers through secure key exchange;
receiving a request to add or delete a node from a load balancer that controls distribution of requests to the group controllers;

generating a new group session key for each node in each branch of the tree that is affected by adding or deleting the node from the group;

distributing the group session key from the first group controller to the other group controllers over the secure channel.

22. A method as recited in Claim 21, further comprising the steps of generating the group session key only when the first group controller is designated as a master group controller that is authorized to join nodes and generate group session keys.

23. A method as recited in Claim 22, further comprising the steps of successively designating different ones of the group controllers as the master group controller in real time.

24. A method for creating a secure multicast or broadcast group, the method comprising the steps of:

establishing a secure communication channel among a plurality of group controllers via a public key exchange protocol;

load balancing traffic emanating from a plurality of nodes to the plurality of group controllers; and

distributing a group session key by one of the group controllers based upon a logical arrangement of the nodes in a binary tree structure, the binary tree structure having a root node, intermediate nodes, and leaf nodes, wherein the plurality of nodes correspond to leaf nodes of the binary tree structure and the group controllers correspond to the root node.

25. The method as recited in Claim 24, wherein the step of distributing further comprises:

circulating a token among the plurality of group controllers to designate the one group controller as having permission to selectively generate the

group session key and keys associated with the intermediate nodes and the leaf nodes; and

selectively generating the group session key based upon the circulating step.

26. The method as recited in Claim 24, wherein the step of distributing further comprises:

detecting whether the secure multicast or broadcast group has a node that is leaving the secure multicast or broadcast group;

determining which of the intermediate nodes are affected in response to the detecting step;

updating keys associated with the affected intermediate nodes;

generating a new group session key; and

sending the new group session key to the leaf nodes.

27. The method as recited in Claim 26, wherein the step of updating comprises:

generating a new key of a parent node of the leaving node; and

encrypting the new key of the parent node with a key of a node adjacent to the parent node.

28. The method as recited in Claim 24, wherein the step of distributing further comprises:

receiving a request message from one of the plurality of nodes to join the secure multicast or broadcast group;

determining which of the intermediate nodes are affected in response to the receiving step;

updating keys associated with the affected intermediate nodes;

generating a new group session key and a private key of the joining node; and

sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

29. The method as recited in Claim 28, wherein the step of updating comprises performing a one way hash function on the keys associated with the affected intermediate nodes.

30. The method as recited in Claim 24, further comprising addressing the plurality of group controllers using a single virtual address.

31. A computer system that can manage addition and deletion of network nodes from and to a secure multicast or broadcast group of nodes in a communications network without a single point of failure, wherein each of the nodes is associated with one of a plurality of replicated group controllers and wherein the nodes and the group controllers are logically organized in a binary tree that represents the network nodes and the group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the group, intermediate nodes represent other network nodes, and root nodes represent the group controllers, the computer system comprising:

a load balancer coupled to the group controllers for interfacing inbound service requests to a selected one of the group controllers;

a bus coupled to the load balancer for transferring data;

one or more processors coupled to the bus for selectively generating a group session key under control of program instructions;

a memory coupled to the one or more processors via the bus;

one or more sequences of program instructions stored in the memory which, when executed by the one or more processors cause the one or more processors to perform the steps of:

joining one of the group controllers to the plurality of replicated group controllers in a local network;

establishing, by one of the group controllers, a secure communication channel between one of the group controllers and another of the group controllers using a key exchange protocol;

receiving a request to add or delete a node of the group from a load balancer that is coupled to the plurality of group controllers;

creating and storing a new group session key for each node in each branch of the tree that is affected by adding or deleting the node from the group;

distributing a group session key from one of the group controllers to the network nodes.